Introduction to Digital Computers / Computer Structure - Spring 2003
Assignment No. 3

Firm Deadline: March 20th - before the beginning of the lecture
EE students: submit questions 1-3.
CE students: submit questions 1-5.

1. Consider the following recursive design called Decoder'(n):

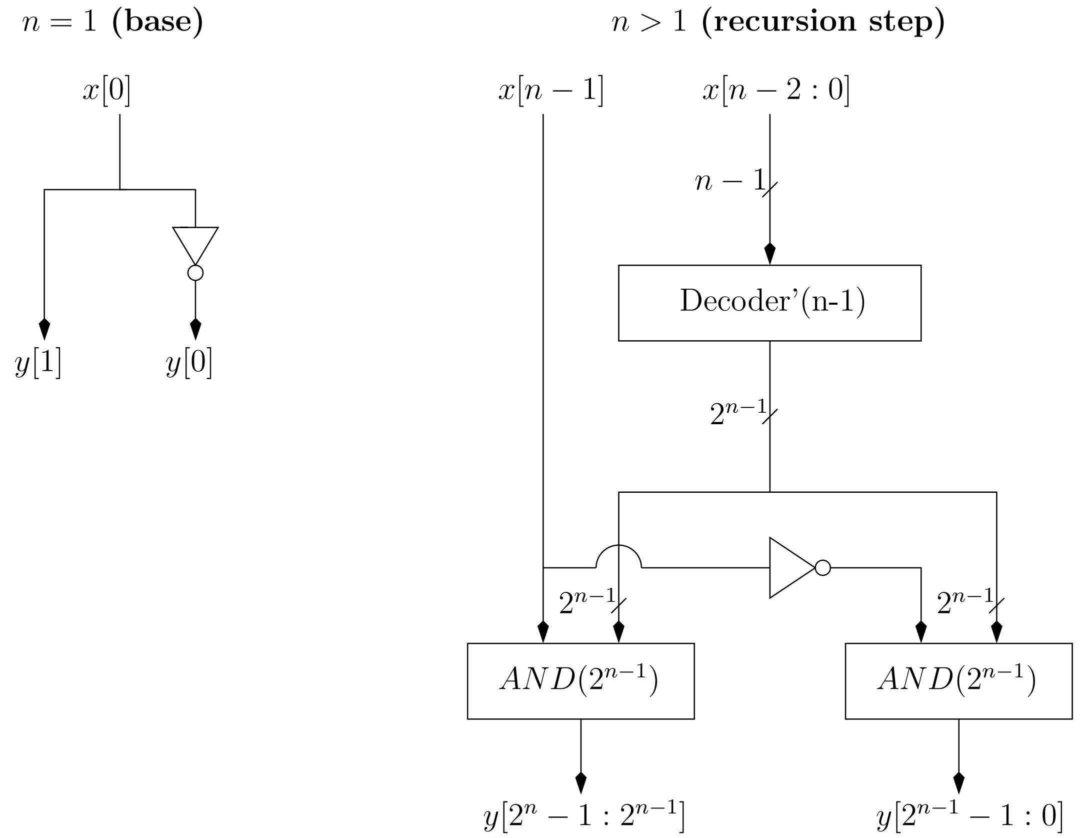

Figure 1: Decoder of n bits

   (a) Prove that the Decoder'(n) design is a correct implementation of a decoder.

   (b) Analyze the cost and delay of Decoder'(n) as a function of n. Write recurrence equations for the cost and delay of Decoder'(n) and solve the asymptotics. Solve the recurrence for the Motorola "technology" for $n = 1, 2, 4, 8, 16$.

2. Consider the decoder depicted in Figure 2 (this is the decoder design presented in class for $k = n/2$). Assume that n is a power of 2.

   (a) Let $I[n-1:0]$ denote the binary representation of i (i.e. $\langle I[n-1:0]\rangle = i$). Write an expression for $A[\text{mod}(i, 2^{n/2})]$ and $B[\lfloor i/2^{n/2} \rfloor]$ in terms of the string $I[n-1:0]$.

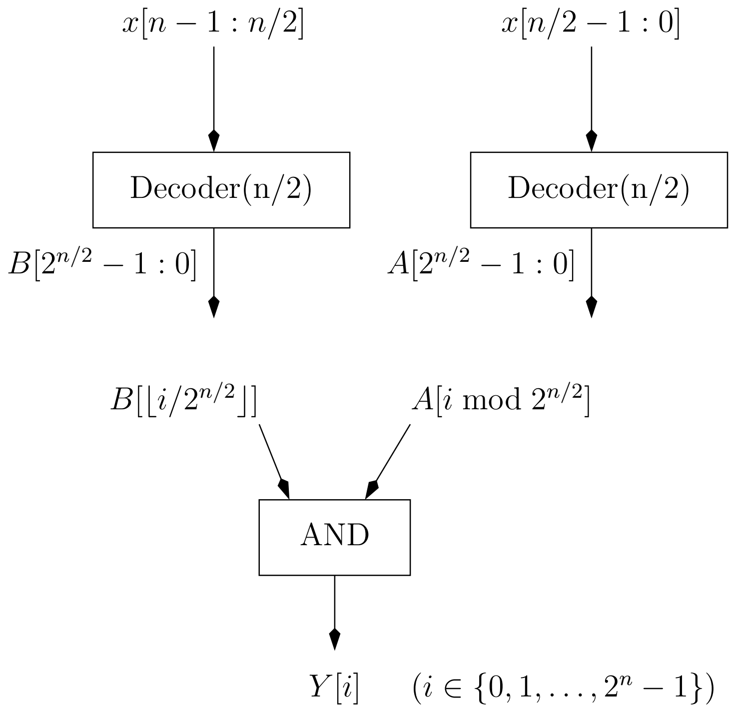

Figure 2: Decoder of n bits

(b) What is the maximum fanout of a net in the decoder as a function of n?

(c) Solve the recurrence equations for the delay and cost of the decoder design for the Motorola "technology" for $n = 1, 2, 4, 8, 16$. Compare with the results of Question 1(b).

3. (a) Assume that n is a power of 2. Prove that if an OR-tree(n) is constructed recursively by using balanced partitions, then the delay of the tree is minimized.

   (b) Consider the following recursive algorithm for computing an OR-tree(n) (for an arbitrary positive integer n):

      i. The case that $n \leq 2$ is trivial.

      ii. If $n > 2$, then let a, b be *any* pair of positive integers such that (i) $n = a + b$ and (ii) $\max\{\lceil \log_2 a \rceil, \lceil \log_2 b \rceil\} \leq \lceil \log_2 n \rceil - 1$.

      iii. Compute an OR-tree(a) and an OR-tree(b). Connect their outputs to an OR-gate to obtain an OR-tree(n).

   Prove that this algorithm computes a minimum delay OR-tree.

   NOTE: Anything you have already proven (correctly!) or proven in class does not need to be proven again. Quoting the argument will suffice.

4. (You may rely on the previous question when answering this question.) The *fanout* of a net is the number of gate inputs that are fed by the net. In many technologies the fanout is limited. In such cases, a gate-output that needs to be fed to many gate-inputs

must pass through a "buffer" that deals with amplifying and restoring the voltage. A d-buffer is a gate that has 1 input and d outputs, and all the output values equal the input value.

Suppose that the fanout is limited by 2, and that one may use 1-buffers the cost of which equals 1 and the delay of which equals 1.

Suppose that a gate-output has to be fed to n gate-inputs. Suggest a way to distribute the signal using 1-buffers so that delay is minimized. Prove that your suggestion is optimal. Can the number of buffers be minimized?

5. Suppose that the fanout is limited by 2, and that one may use 1-buffers the cost of which equals 1 and the delay of which equals 1.

    (a) Rewrite the recurrence equations for the delay and cost of the two types of decoders described in Question 1 and Question 2. (Assume that n is a power of 2)

    (b) Compute the cost and delays according to the Motorola technology of the two types of decoders for $2 \leq 2^n \leq 128$ with and without the fanout constraint.